A TECHNIQUE FOR SCHEDULING SUBCONTRACTED WORK IN A CYCLIC PROJECT

A THESIS

Presented to

The Faculty of the Division of Graduate

Studies and Research

by

Samuel Marion Burney, Jr.

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in the School of Industrial and Systems Engineering

Georgia Institute of Technology

June, 1971

A TECHNIQUE FOR SCHEDULING SUBCONTRACTED WORK

IN A CYCLIC PROJECT

Approved:

Chairman

Date approved by Chairman: June 4, 1971

ACKNOWLEDGMENTS

The author would like to express sincere appreciation to his thesis advisor, Dr. J. G. Davis, for his invaluable help in the preparation of this thesis. Dr. Davis provided initial impetus to the research and gave much valuable time and effort to the project. The consideration given by the members of the reading committee, Dr. J. J. Jarvis and Dr. R. B. Cooper, was also very much appreciated.

Last, but certainly not least, the author's wife, Sandy, receives thanks for her help in preparing the various drafts of the paper. Her patience and understanding throughout the project were instrumental in its completion.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Concluded)

SUMMARY

This study presents a technique for reducing work interruptions for a single activity in a cyclic project. The precedence relationships of the project preclude the performance of all cyclic activities on a continuous basis.

Work delays will often cause additional costs, since they may result in idle man-days, increased costs of work set-up, or a decrease in the effect of the learning factor. It is, therefore, desirable to promote economy of crew utilization.

It is particularly desirable to promote work continuity for a subcontracted activity. Subcontracted crews are normally so specialized that they may not be utilized elsewhere when idle. Consequently, the technique provided for promoting work continuity is particularly applicable for use with a subcontracted activity.

This technique allows one to determine the number of work interruptions necessary for an activity without requiring the development of a detailed work schedule. It thus provides information which would be useful in the project planning phase without requiring costly network scheduling. This information could be used as a basis for negotiating contracts with potential subcontractors or as a means of determining the cost effectiveness of various scheduling alternatives.

CHAPTER I

INTRODUCTION

General Background Information

Within the past fifteen years there have been many developments in the area of network planning. The discovery of the CPM and PERT techniques seemed to trigger a virtual explosion of work in the area.

The Critical Path Method (CPM) was developed by James E. Kelley and Morgan R. Walker for E. I. duPont de Nemours and Company. It was first applied on a large scale in 1957. It is an activity oriented method based on the premise that a project can be shortened by applying extra cost. The method produces least cost work schedules for each of several project durations and calculates that project duration which gives least project cost.

The Program Evaluation and Review Technique (PERT) originated in the Special Projects Office of the Department of the Navy. It was first implemented on the Polaris missile program in 1958. Whereas CPM was activity oriented, this technique is event oriented -- the results of PERT analysis are expressed in terms of events, or steps of progress. Multiple time estimates (indicating time uncertainty) for activities are used in calculating the probability of achieving scheduled completion times.

From these two basic systems, numerous network planning systems have been developed. All of these systems fall into one of three general categories:

1. systems dealing only with time,

2. systems which analyze cost and time for the case of unlimited resources, and

3. systems which consider resource availability in conjunction with costs and produce cost-optimized work schedules within preset resource limitations.*

The network scheduling systems currently available are designed primarily for use with nonrepetitive networks. As such, they do not always provide satisfactory solutions when used on a project which has repetitive activities. Such a project will be referred to as a cyclic project and will be described in more detail below.

Work interruption can be a serious problem in a cyclic project. Because of precedence requirements, it is frequently necessary to delay work on a particular activity until its predecessor activities within the same cycle have been completed. These delays cause inefficiency for a variety of reasons: (1) setup costs may be incurred again; (2) the effect of the learning factor is diminished at the very least; (3) delays may cause lost man-days if the crews cannot be utilized elsewhere; and (4) there may be increased administrative costs if it is necessary to fire a crew after completion of a cycle and rehire later.

At the present time, there is no system which will minimize work

---

*H. S. Woodgate, Planning By Network, p. 24.

interruptions for specific activities in a cyclic project. A maximum continuity schedule of this type might be very desirable when dealing with certain types of activities.

In general, any activities whose interruption would cause a significant loss in efficiency (for any of the reasons listed above) would be good examples of the type of activities for which a maximum continuity schedule might be desirable. In particular, it would be advantageous to derive a maximum continuity schedule for a subcontracted activity since it might prove helpful in negotiating a contract. Throughout this paper, the reduction of interruptions for a subcontracted activity will be discussed since it provides a convenient manner for referring to the activity in question and it is a type of activity for which the system may be particularly useful. It should be remembered, however, that non-subcontracted activities may also benefit from maximum continuity scheduling.

This study will attempt to do several things. First, it will point out the need for a system which will minimize work interruptions on a subcontracted activity. Second, it provides a method of reducing work interruption for a specific subcontracted activity (while still completing the project in the minimum amount of time). And finally, it describes the possibilities for extending the method to handle multiple subcontracted activities, the problems which would be involved, and the various costs which may be incurred as a result of reducing work interruption.

Before going into a description of the problem itself, it may be helpful to insure that the principal terms are understood. Provided on the following pages are explanations as to just what is meant by a "cyclic project" and "subcontracted work."

Description of a Cyclic Project

A cyclic project is one which requires the repetition of distinctive phases of work. A good example might be the construction of an apartment building. In this example, the project would be the construction of the entire building, and the cyclic phases would be the construction of the individual apartments. Virtually the same work must be done in constructing each apartment throughout the building.

It should be noted at this time that the primary objective in the scheduling of a cyclic project is normally the same as that for any other project -- complete the project by a certain time. In the example cited above, this would mean that the apartment building should be completed by a certain time. In some projects, it may be necessary to complete the individual cycles as soon as possible also. Again referring to the above example, this requirement might be made if the owner wished to move people into some apartments while work was being completed on the others.

This type of requirement represents a special case, however. Generally, we will require only that the project itself be completed by a specified date and make no special requirements as to when individual cycles be completed.

In the absence of specific project cost data, it is best to level resource usage as a means of minimizing project cost. Moder and Phillips (18) describe a method by which the resources are leveled within the cyclic portion of the schedule without regard to precedence requirements. The length of the longest activity sets the length of individual cycle duration. Precedence requirements are satisfied by the addition of activities before and after the cyclic portion of the schedule.

As an example, consider the activity sequence shown in Figure 1 to represent the steps required in constructing a "widget." If our project is to construct a large number of widgets, we would have a cyclic project. In this particular project, the cycle length would be four days, since activity 3-5 has the longest duration and its length is four days. In attempting to minimize the cost of this project, we would arrange all of the activities in a four day cycle without regard to precedence relationships so as to best level resource requirements.

For this project, a leveled cycle would look like that shown in Figure 2. Once the cycle has been determined, we should work back to the start in order to satisfy precedence relationships. The resulting steps will be referred to as the set-up portion of the network diagram. Once this has been completed, we can move to the last cycle and add sufficient activities to insure that all widgets which have been started are completed. This will be referred to as the "shutdown" phase of the network.

Figure 3 represents the network diagram which might be used in a project requiring the construction of n + 5 widgets. Notice that three widgets are begun in the set-up phase and one in each of the n + 2 cycles. Thus, a total of n + 5 are produced.

This schedule represents a satisfactory solution if nine resource units are available for the project. Obviously, new problems would be introduced if we did not have nine resource units available. The project length would have to be increased in order to meet such a limitation on resources.

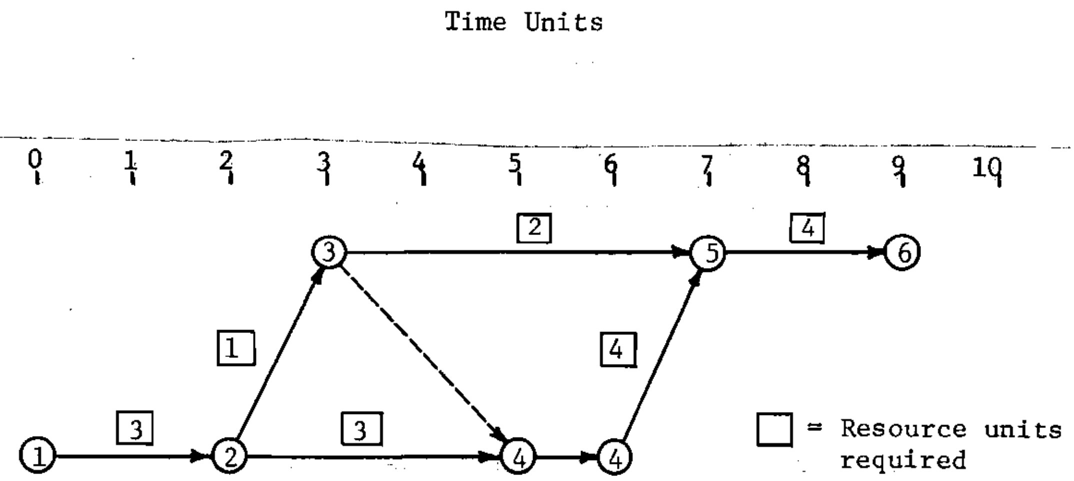


Figure 1. A Time Scaled Network Diagram for One Cycle of the Project

Time Units

Activity	1	2	3	4
1-2	3	3		
2-3		1		
2-4		3	3	3
3-4				
3-5	2	2	2	2
4-5	4			
5-6			4	4
Total Resources:	9	9	9	9

One Cycle

Figure 2. A Cycle with Leveled Resources

Setup | 1st Cycle | 2d Cycle | n Cycles | Shutdown

Duration		Resource Requirements	Activity Number		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	. . .	. . .	18 +4n	19 +4n	20 +4n	21 +4n	22 +4n	23 +4n	24 +4n
2		3	1-2		3	3	3	3		3	3			3	3			3	3												
1		1	2-3				1				1				1				1						1						
3		3	2-4				3	3	3	3	3	3			3	3	3		3	3	3				3	3	3				
0		0	3-4																												
4		2	3-5					2	2	2	2	2	2	2	2	2	2	2	2	2	2			2	2	2	2	2	2		
1		4	4-5							4				4				4						4					4		
2		4	5-6									4	4			4	4			4	4						4	4		4	4
Total resource requirements					3	3	7	8	8	9	9	9	6	9	9	9	9	9	9	9	9	9	9	6	6	5	9	6	6	4	4

Figure 3. A Project Schedule with Leveled Resources

Description of Subcontracted Work

Subcontracted work is very often a part of large scale projects. A construction firm, for example, would normally subcontract for electrical and plumbing work required in the construction of an apartment complex.

The usual procedure which is followed in letting a subcontract is to specify the amount and type of work to be done and receive bids from firms interested in performing the work. In specifying the work to be done, it is not common practice to indicate the day-by-day schedule of when this work is to be done. Those parties submitting bids must, therefore, make assumptions regarding the possible work schedule in preparing their bid. As mentioned previously, work delays might result in considerable inefficiency for a subcontractor. In the absence of scheduling information, the subcontractor must determine his bid with the understanding that numerous interruptions may be encountered. Obviously, he could submit a more precise bid if he had some idea as to the probable number of interruptions to be expected.

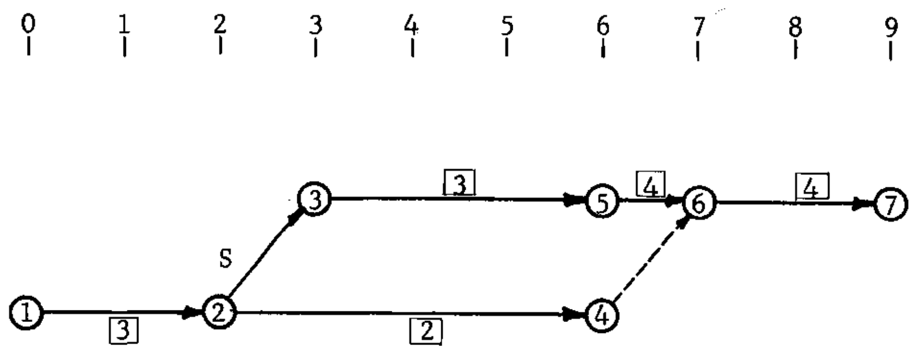
Description of the Problem

On far too many projects, subcontracted work is treated as a sort of buffer in network scheduling. It is scheduled in a manner so as to enhance leveling of the non-subcontracted resources with little regard given to interruption of the subcontracted work. In other words, the total subcontracted job is treated by the general contractor as a fixed cost quantity no matter how it is scheduled.

As mentioned previously, there is considerable inefficiency inherent in this type of operation. If the subcontractor knew the number of work interruptions required in his project schedule (or if he had an idea of the maximum number of work delays), he might be able to submit a lower bid on the project because his total costs would be less if there were fewer interruptions on his schedule.

In most cases, a schedule designed to minimize work interruptions in a given class of activities will result in some type of cost increase in the remainder of the schedule -- for example, the resource leveling of other activities may be adversely affected. Consider the ten cycle project schedules on the following pages. Figure 5 represents a schedule with the best resource leveling attainable for the network shown in Figure 4. Figure 6 is a schedule which minimizes the number of work interruptions of the subcontracted activity 2-3 for the same project. In this case, the number of interruptions has been reduced from 3 to 1, but the sum of squares of daily resource levels required for the non-subcontracted activities (an indicator of relative leveling as described by Burgess and Killebrew (3)) has been increased from 2918 to 2972. Thus, we have reduced work interruptions while increasing the variability in resource levels.

If the subcontractor knew that he would have a schedule calling for only one work interruption, he might submit a lower bid than he would if he felt there was a possibility of 10 interruptions, and the savings resulting from a lower bid might more than offset the costs incurred in implementing this schedule. In a case where the project manager was not willing to commit himself to the schedule shown in Figure 6, he might request that bids be submitted which are functions of the number of interruptions. Such bids would provide him with knowledge of the cost of work



□ = daily resource usage

S = subcontracted

Figure 4. Time Scaled Diagram of One Cycle of a Ten Cycle Project

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1-2	3	3	3	3	3	3	3	3			3	3			3	3			3	3		
2-3			S				S	S	S										S	S	S	
2-4			2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
3-5				3	3	3		3	3	3		3	3	3		3	3	3		3	3	3
5-6							4				4				4				4			
6-7									4	4			4	4			4	4			4	4
Daily Totals	3	3	5	8	8	8	9	8	9	9	9	8	9	9	9	8	9	9	9	8	9	9

	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44
1-2	3	3			3	3			3	3												
2-3									S	S	S											
2-4	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2		
3-5		3	3	3		3	3	3		3	3	3		3	3	3	3	3	3			
5-6	4				4				4				4				4			4		
6-7			4	4			4	4			4	4			4	4		4	4		4	4
Daily Totals	9	8	9	9	9	8	9	9	9	8	9	9	6	5	9	9	9	9	9	6	4	4

Figure 5. A Schedule with Leveled Resources for the Network Shown in Figure 4

					5					10					15					20		
1-2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3		
2-3			S										S	S	S	S	S	S	S	S	S	
2-4			2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
3-5				3	3	3								3	3	3		3	3	3	3	3
5-6							4										4				4	
6-7									4	4								4	4			4
	3	3	5	8	8	8	9	9	9	5	5	5	5	8	8	8	9	12	12	8	9	9

			25					30					35					40					45
1-2																							
2-3																							
2-4	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2			
3-5	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3				
5-6		4			4			4			4			4			4			4			
6-7	4		4	4		4	4		4	4		4	4		4	4		4	4		4	4	
	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	6	4	4	

Figure 6. A Schedule Allowing for Maximum Continuity of the Subcontracted Work

interruption. This information would allow the manager to determine the cost effectiveness of reducing interruptions.

Assumptions Made

The following assumptions were made in working on this problem:

1. The cost of work interruption is the same regardless of where the interruption occurs. It is no more desirable to have a delay following the first cycle than it is following the n^{th}.

2. No duplicate crews are available for work on the project. It is, therefore, not possible to be working on more than one cycle of an activity at a particular time.

3. The primary objective is to complete the entire project in the minimum amount of time.

Purpose of the Research

Although there has been a great deal of work done in the area of network scheduling, little consideration has been given to the peculiar problems inherent in the scheduling of cyclic projects. Consequently, there is much work to be done in this area.

Basically, the aim of this study is to provide groundwork for future work in this area. As such, the purpose is:

1. to emphasize the need for network scheduling systems which maximize work continuity;

2. to develop a system which maximizes work continuity for a single activity in a cyclic project;

3. to point out possible extensions of the network scheduling system developed; and

4. to discuss the ramifications of maximum continuity scheduling.

A Review of Literature Relating to This Subject

There has been very little written about the scheduling of cyclic projects. Papers by Fisher and Nemhauser (9) and Burgess and Killebrew (3) deal with the problem, and there is some mention of multicycle scheduling in a book by Moder and Phillips (18).

Several books provide good textbook-type treatment of basic network planning techniques. Those by Moder and Phillips (18), Woodgate (29), and Cleland (5) are examples of these.

In all of the literature surveyed, there was no mention of attempts to maximize work continuity in a cyclic project. There are work interruption penalties in some scheduling algorithms, but these penalties are provided for the interruption of a particular activity, not for the separation of cyclic activities. It is, therefore, a completely different problem. The problem with which this paper deals has been completely ignored in the literature to this date.

Much work has been done in the field of resource allocation. Since subcontracted work might, in some instances, be treated as an activity utilizing a limited resource, it might prove helpful to review those resource allocation procedures currently available.

Papers by Carruthers and Battersby (4), E. W. Davis (7), and Rosenbloom (23) provide a good review of the work done in the area of resource allocation. Resource allocation procedures may be generally classified into two major categories -- mathematical programming models and heuristic programming models.

Mathematical Programming Models

There is evidence of considerable effort being made in applying mathematical programming to resource allocation. The technique of branch and bound was applied by Johnson (11) with good results for projects with fewer than fifty activities. The computer time required for solution becomes excessive, however, beyond that point.

Several attempts have been made at applying a zero-one integer programming technique since resource allocation is basically a combinatorial problem. Pritzker and Watters (21,22) and Wolfe (28) have shown the superiority of their methods over known heuristics for simple theoretical examples. Similar methods used by Meyer and Shaffer (17) and Crowstone and Thompson (6) have, however, proven to be computationally infeasible for large practical problems.

Another integer programming approach is provided by Bennet (1). He developed two linear models on which an integer programming approach was tried. By his own admission, however, the approach would not be feasible in the planning of real-world projects because of limited computer memory capacity and excessive run time.

A dynamic programming approach to the resource allocation problem was attempted by Petrovic (20). He maps the predecessor-successor relationships of activities into a set of transformations where the state variables are the amounts of work required for each activity. The resources allocated to each activity are viewed as decision variables. It is an attempt to formulate a precise mathematical model and, as such, is rather detailed for practical use.

Another approach was taken by Jewell (10). His technique is restricted to the case of assumed continuous, convex activity time-resource functions. He assumes that the planner must set up a fixed project schedule in the face of uncertain activity durations. Based on the difference between the initially allotted time and the actual time needed by the activity, additional resources may be required to meet the schedule. He then addressed the problem of how to schedule the project so as to minimize the amount of additional resources required. Quadratic programming is used in accomplishing this. Computer time becomes a problem for large projects.

Another approach for the restricted case of continuous convex activity time-cost functions is offered by Berman (2). He describes a model which allocates resources in a PERT network, the activities of which are subject to continuous, concave-upward, time-cost functions, in such a way as to achieve a minimum cost solution for a particular completion date. Essentially, his procedure is to analyze the network in terms of "junctions" (or events). He fixes all but one and minimizes cost by manipulation of that one. Understandably, the procedure is not optimal.

Heuristic Programming Models

Davis (7) indicates that there are two types of resource allocation problems: smoothing resource usage if sufficient resources are available, or the scheduling of activities to meet limitations when resource availability is restricted.

A systematic approach to the problem of leveling resources is provided by Burgess and Killebrew (3). They use the sum of squares of the resource requirements as a measure of effectiveness. This measure has the property of becoming smaller as the variation in resource requirements from time-unit to time-unit becomes smaller.

Levy, Thompson, and Wiest (14) describe a computer program for smoothing manpower requirements which is similar in many respects to the method described previously. It is, however, designed to handle several projects simultaneously. Wilson (27) produced a slightly different version of the Levy procedure. Simplifying assumptions were made in developing this scheme and, consequently, it is considerably less flexible than the Levy method. For large problems, it would lead to computational complexities.

As mentioned previously, the second type of resource allocation problem is that of determining the minimum project duration which can be achieved when available resources are subject to stated constraints. Obviously, the above mentioned procedures would be applicable for this type of problem if the limits on resource availability were set high enough.

Wiest (24,25,26) developed a heuristic model for scheduling large projects with limited resources. The model lists the currently available activities, for any given period, in the order of the amounts of total slack available. In this way, he insures that the most critical activities have the highest probability of being scheduled first. The activities are then scheduled up to the resource limit using Monte Carlo techniques, and unscheduled jobs are pushed forward.

Several computerized methods have been developed which deal with the problem of scheduling with limited resources. The RAMPS (30) method is probably the most widely used. It was developed by C.E.I.R. -- a computer service organization. It is designed to handle several projects simultaneously and schedule each activity so that project completion dates are met, and idle resources are minimized subject to stated constraints. Since it is a proprietary algorithm, details of the technique are not known. From the descriptive information provided, however, we may deduce that it is a heuristic technique which performs pointwise optimization. Klein (12), Lambourn (13), and Moshman, Johnson, and Larsen (19) have authored papers on the RAMPS technique. Each provides some insight on the technique itself without going into details of the system.

Martino (16) has developed another technique referred to as the Multiple Resources Allocation Procedure (MAP). The procedure, basically, consists of eight rules for allocating resources. The first rule gives a simple formula for estimating the most likely crew size and the following rules provide a step-by-step procedure for determining if that crew size is indeed optimal. If not, measures are taken to adjust it. The procedure becomes tedious for large projects as one might imagine.

Another procedure was developed by McNeill (15) and modified by Davis and Buchan (8). This technique is referred to as the REST Algorithm. It treats activities with variable resource and time requirements and requires only a single pass in order to reach a solution. It does, however, require rather detailed estimation of the effects of varying crew sizes on an activity's duration.

The state-of-the-art in resource allocation techniques is summarized well by Carruthers and Battersby (4) as follows:

No algorithms for the resource smoothing problem exist which are rigorous, general and practicable; some exist which are rigorous, or rigorous and practicable, but only in special cases. The rigorous approach to resource smoothing considers the entire project as one entity, and each decision in scheduling is made against the global requirements of the project. However, all practicable algorithms are intended to be reasonable rather than optimal, with regard to an objective function; consequently they are based on local decisions. Resource smoothing is currently a process of "melioration" rather than "optimization."

CHAPTER II

PROCEDURE

The procedure followed in developing a system to maximize work continuity for a subcontracted activity in a cyclic project was more tedious than difficult. It was first necessary to determine which network factors influenced the scheduling of the subcontracted activity.

In order to understand the importance of certain network factors in determining a maximum continuity work schedule, it is necessary to have some understanding of the basic procedure followed in maximizing continuity for a particular activity. Keep in mind the fact that there are three basic requirements in deriving any schedule.

1. Finish the project in a minimum amount of time.
2. Satisfy all precedence relationships.
3. Based on an assumption of no duplicate crews, no work may begin on a particular activity in cycle $n + 1$ until that same activity has been completed in the n^{th} cycle.

Because of this third requirement, it is rather obvious that the total project length will never be less than the duration of the longest activity in the network (the pacing activity) times the total number of cycles to be run.

In order to minimize project length, it is necessary to begin work on the pacing activity on its latest start and to perform all cycles of that activity without interruption. Predecessor activities must be scheduled in some manner so that their precedence relationships with the pacing activity <u>and</u> with each other are satisfied.

The schedule for the subcontracted activity has now been constrained in one direction. If this activity precedes the pacing activity in the network, a particular cycle of the subcontracted activity must be completed by a certain time in order that the corresponding cycle of the pacing activity is not delayed. Similarly, if the subcontracted activity follows the pacing activity, a cycle may not be started until the corresponding cycle of the pacing activity <u>and</u> any intermediate activities have been completed. The length of the pacing activity is, therefore, a very important network factor.

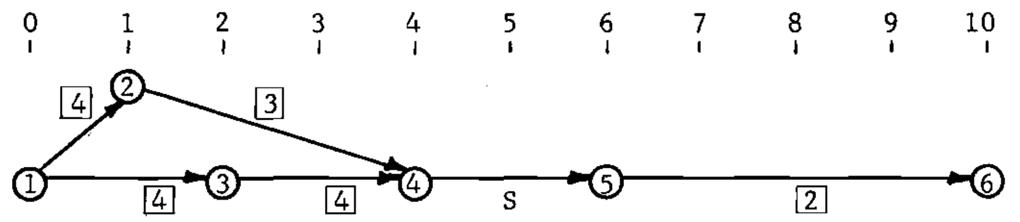
It so happens that the longest predecessor or successor to the subcontracted activity (depending on whether the pacing activity succeeds or precedes the subcontracted activity) constrains the subcontracted schedule also. This results from the fact that continuity of the subcontracted schedule is promoted by delaying the start of work as long as possible and then performing as many cycles as precedence relationships will allow.

In the case where the pacing activity precedes the subcontracted activity, the amount by which the start of the subcontracted activity can be delayed will be determined by the amount which the longest successor activity can be delayed since they are precedence related. The subcontracted activity's start may never be delayed so much that the corresponding cycle of the longest successor is delayed beyond its latest start. Hence, a particular cycle of the subcontracted activity must be completed

X time units before the latest start of the corresponding cycle of the longest successor activity. The X represents the length of the longest path in a cycle between latest completion of the subcontracted activity and start of the longest successor. In a single path network, this would be the sum of the durations of intermediate activities. Similarly, the longest predecessor provides a constraint in the case where the pacing activity follows the subcontracted activity in the network.

So far, three factors have been deemed important: the length of the pacing activity; the length of the longest predecessor (or successor) activity; and the longest path from latest completion of the subcontracted activity to start of the longest successor. In the case where the subcontracted activity precedes the pacing activity, the longest paths from completion of the longest predecessor to start of the subcontracted activity and from latest completion of the subcontracted activity to start of the pacing activity become important for reasons that are apparent from the type of constraints which the longest predecessor and the pacing activity place upon the subcontracted activity's schedule.

Finally, the free slack available to the subcontracted activity and its longest predecessor are important factors in those networks where the subcontracted activity and/or the longest predecessor activity are off the critical path. The slack is important since it influences the amount of delay which is possible for the start of a particular cycle of the subcontracted activity. Figures 7 and 8 provide maximum continuity schedules for two networks which are similar in all respects except the amount of slack available to the longest predecessor to the subcontracted activity. In the case where the longest predecessor is off the critical



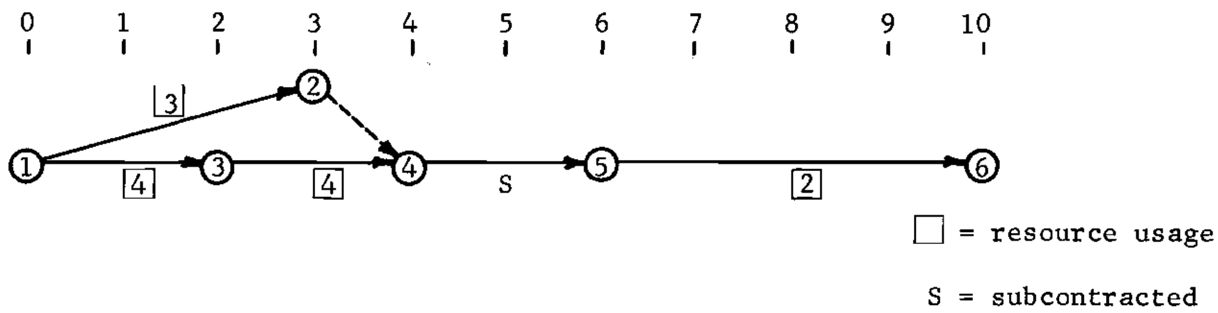
☐ = resource usage

S = subcontracted

					5					10					15					20	
1-2	4	4	4	4	4	4	4	4	4												
1-3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4			
2-4		3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
3-4			4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
4-5					S	S			S	S	S	S					S	S	S	S	S
5-6							2	2	2	2	2	2	2	2	2	2	2	2	2	2	2

				25					30					35					40		
1-2																					
1-3																					
2-4	3	3	3	3	3	3	3														
3-4																					
4-5	S	S	S										S	S	S	S					
5-6	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2

Figure 7. A Schedule Allowing for Maximum Continuity of Activity 4-5 (One Cycle Network Shown)



					5					10					15					20	
1-2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
1-3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4			
3-4			4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
4-5					S	S	S	S					S	S	S	S	S	S	S	S	
5-6							2	2	2	2	2	2	2	2	2	2	2	2	2	2	2

				25					30					35					40		
1-2	3	3	3	3	3	3															
1-3																					
3-4																					
4-5								S	S	S	S	S	S								
5-6	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2

Figure 8. A Schedule Allowing for Maximum Continuity of Activity 4-5 (One Cycle Network Shown)

path (Figure 8), we have only two interruptions of subcontracted work. In the other case, there were three interruptions for the nine cycle project.

Thus, the important network factors are those mentioned above. These were the only factors which were found to influence the determination of the minimum number of work interruptions for a particular subcontracted activity. The fact that any network, regardless of size may be reduced to a description of these factors in determining the number of interruptions necessary for an activity is of considerable importance. In most cases, this allows for the determination of the number of work interruptions without requiring development of the entire project schedule. The project manager could, therefore, determine the number of interruptions (and request bids from subcontractors based on this determination) without having to waste time and money developing detailed schedules which might never be used.

After determining the network factors which affect the scheduling of the subcontracted activity, it was necessary to derive a procedure for maximizing work continuity which would be applicable for all the possible network types with respect to these factors. Thus the work scheduling system would be suitable for use in any cyclic project network.

Work continuity could be maximized through an enumeration procedure. By examining every possible schedule for the project, one could determine the schedule with the fewest number of interruptions. This would be a very lengthy task, however, for networks with more than a few cycles and activities. For any practical problem, such a procedure would not be feasible.

The various types of networks which might be encountered were examined and scheduling techniques were determined for each. After determining the techniques required, the steps followed in developing the scheduling algorithms were similar to those followed in the development of any algorithmic procedure. Cases which could be handled by similar techniques were grouped together and the solution procedures were diagrammed.

CHAPTER III

DISCUSSION OF RESULTS

As was previously mentioned, those network factors which affected the scheduling of the subcontracted activity were determined in order that some general type of scheduling procedure could be devised. The factors which were found to be important are summarized below.

1. The length of the pacing activity.

2. The length of the subcontracted activity.

3. The length of the longest activity which has a precedence relationship with the subcontracted activity which is the opposite of the relationship between the pacing activity and subcontracted activity.

4. The precedence relationship between the subcontracted activity and the pacing activity.

5. The free slack available to the subcontracted activity and the longest predecessor to the subcontracted activity.

6. a. For networks in which the subcontracted activity precedes the pacing activity, the length of the longest path from completion of the longest predecessor (to the subcontracted activity) to start of the pacing activity in a particular cycle.

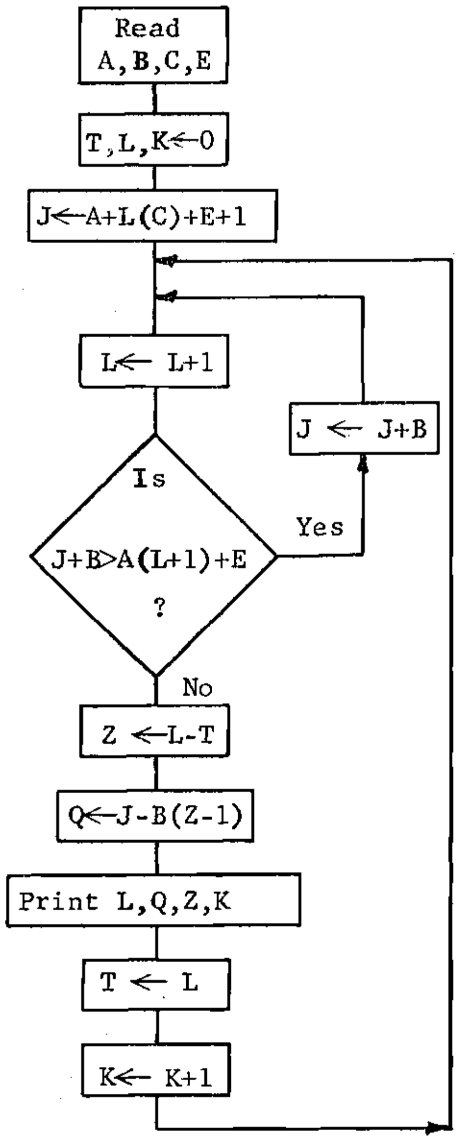
b. For networks in which the pacing activity precedes the subcontracted activity, the length of the longest path from completion of the pacing activity to the start of the longest successor (to the subcontracted activity) in a particular cycle.

7. The length of the longest path from completion of the longest predecessor activity until start of the subcontracted activity in a cycle.

It should be realized that factors 6 and 7 might be stated in many different ways. Basically, what is required is knowledge as to the time required to perform those activities which occur between the pacing activity and the subcontracted activity and the time required to perform the activities occurring between the subcontracted activity and its longest predecessor (or successor). Factors 2, 6, and 7 provide this information when considered together.

Once these factors had been determined, it was necessary to develop scheduling systems which would handle any type of network. It was found that some generalizations could be made, but not enough so as to handle all cases with one algorithm.

In order to handle every conceivable combination of the above factors, it was necessary to develop two algorithms and two procedural methods. These four scheduling procedures are included as Figures 9, 10, 11, and 12. The first algorithm is designed to handle all networks in which the subcontracted activity precedes the pacing activity. The second algorithm may be used for all networks in which the subcontracted activity is a successor to the pacing activity and is not longer in duration than all of its successor activities. The first procedural method handles the case where the pacing activity precedes the subcontracted activity and the subcontracted activity is longer than all its successor activities. The second procedural method will handle all networks in



Key:

Input

A = duration of longest predecessor to the subcontracted activity

B = duration of subcontracted activity

C = duration of pacing activity

E = time from EC of longest predecessor until LS of the subcontracted activity in a cycle

Output

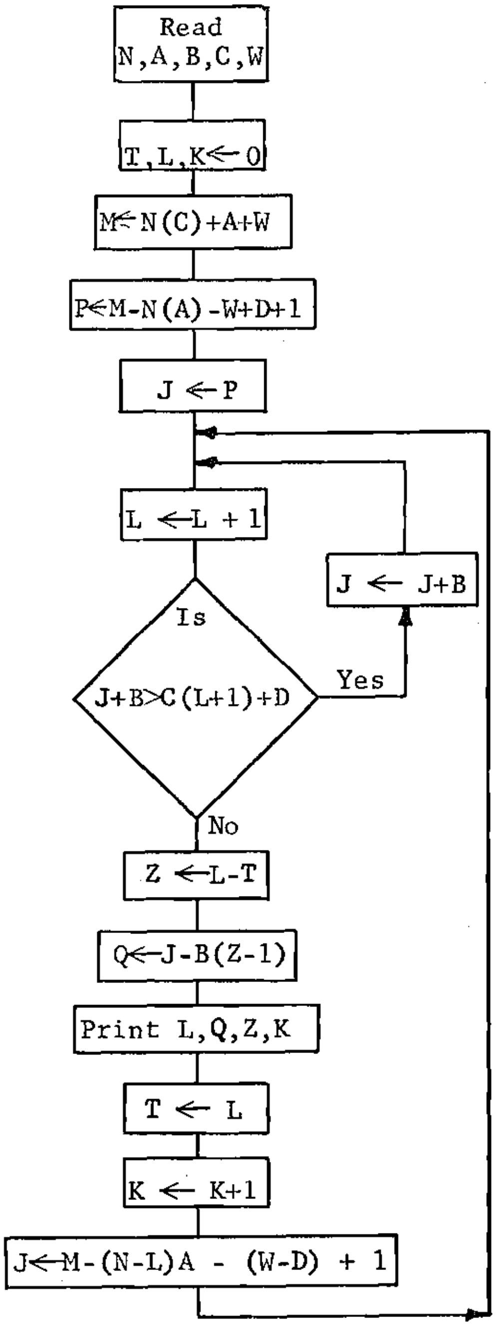
Q = day on which subcontracted work begins measured from initial start of longest predecessor

Z = number of cycles of subcontracted activity performed before interruption

L = total number of cycles performed with K interrupts

K = number of interrupts for L cycles

Figure 9. Algorithm for Reducing Work Interruptions in Networks Where the Subcontracted Activity Is a Predecessor to the Pacing Activity



Key:

<u>Input</u>

N = number of cycles desired

A = duration of longest successor to subcontracted activity

B = duration of subcontracted activity

C = duration of pacing activity

W = time from EC of pacing activity to LS of longest successor to subcontracted activity in a cycle

D = time from EC of pacing activity to LS of subcontracted activity in a cycle

<u>Output</u>

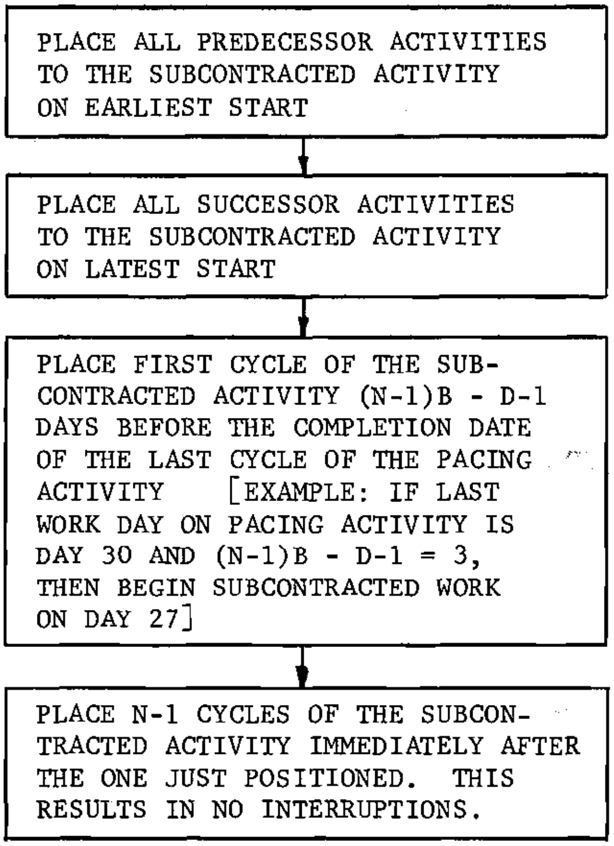
Q = day on which subcontracted work begins

Z = number of cycles of the subcontracted activity performed (beginning at day Q) before an interruption

L = total number of cycles performed with K interruptions

K = number of interruptions for L cycles of the subcontracted activity

Figure 10. Algorithm for Reducing Work Interruptions in Networks Where the Pacing Activity Precedes the Subcontracted Activity and the Subcontracted Activity Is Not Longer in Duration Than Each of Its Successors



KEY:

N = number of cycles desired

B = duration of the subcontracted activity

D = longest path from completion of the pacing activity to start of the subcontracted activity in a cycle

Figure 11. Procedure for Minimizing Work Interruptions in Networks Where the Pacing Activity Precedes the Subcontracted Activity and the Subcontracted Activity Has a Longer Duration Than Any of Its Successors

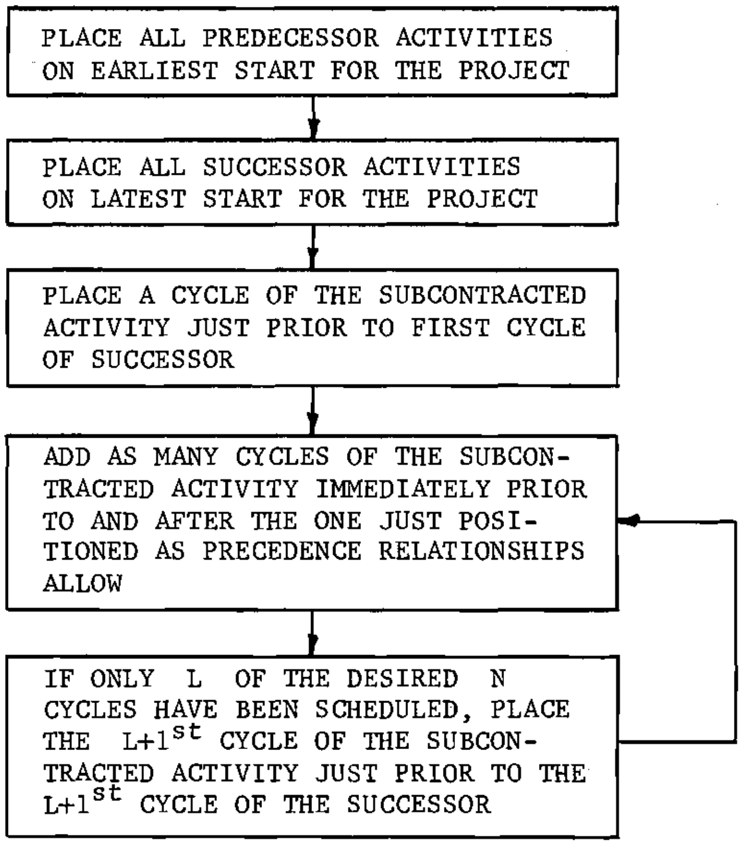


Figure 12. Procedure for Reducing Work Interruptions in Networks Where the Pacing Activity and the Subcontracted Activity Are Not Precedence Related

which the subcontracted activity is not precedence related to the pacing activity.

For the case where the subcontracted activity precedes the pacing activity, the algorithm will place the first cycle of the subcontracted work in a manner so that it will not delay work on the pacing activity. This is done in the step where $J \leftarrow A + L(C) + E + 1$. The first cycle thus begins one time unit after the longest predecessor (to the subcontracted activity) and all intermediate activities (those activities which must occur after the longest predecessor and before the subcontracted activity) have been completed. Free slack for both the longest predecessor and subcontracted activity is added since each allows for delay in the start of the subcontracted activity without any effect on the earliest start of the pacing activity.

After placing the first cycle of the subcontracted activity, the L counter is incremented. This indicates the number of cycles of the subcontracted activity which has been scheduled. The comparison block which follows determines whether precedence requirements will allow another cycle of the subcontracted activity to be added. If it is possible, both J and L are incremented to effect the addition of another cycle and the comparison is again made. It should be noted that the first comparison test made will determine whether the second cycle of the subcontracted activity can begin at $J_o + B$. In order for this to be allowed, its predecessor activities (whose lengths are denoted by $A(L + 1) + D$) must have been completed on or before $J_o + B - 1$. If it were possible to start the second cycle at $J_o + B$, a comparison would then be made to see if the third cycle could logically begin at $J_o + 2B$, etc.

If the comparison shows that the second cycle of the subcontracted activity may not be begun at J + B, the algorithm will provide output as to the number of cycles scheduled since last interruption, the day on which the first of these cycles should be scheduled, the total number of cycles scheduled, and the total number of interruptions. It will then place the next cycle of the subcontracted activity in a position so that it will not delay start of the corresponding cycle of the pacing activity. From that point, the procedure becomes repetitive.

The algorithm which is used in networks where the pacing activity is a predecessor to the subcontracted activity differs from the other in that it is necessary to find the latest start for the longest successor to the subcontracted activity. With the case considered previously, it was known that the pacing activity should be started as soon as possible and this allowed for positioning of the first cycle. In this case, it is possible to delay the longest successor beyond its earliest start. The allowable delay must be determined since delaying of the subcontracted activity promotes continuity and the amount by which the subcontracted activity can be delayed depends on the delay allowable for its successor activities. After the first cycle is positioned, the procedure followed becomes very similar to that used previously. The comparisons made and the incremental steps taken after a delay differ from those of the algorithm described earlier due to the difference in the type of networks handled. The basic idea of the procedure, however, is exactly the same.

It should be realized that the schedule which is derived for the subcontracted activity through the use of these algorithms is not relative to the project start. All activities before the longest predecessor to the subcontracted activity have been ignored in the development of the schedule. The schedule is relative to the start of the longest predecessor activity rather than the start of the project itself. In order to make the schedule derived relative to the start of the project, it is merely necessary to add the earliest start time for the longest predecessor activity to the times in the schedule. The fact that it is possible to ignore network activities is an interesting feature of these algorithms.

The procedural methods found in Figures 11 and 12 require no explanation. They are not so convenient as those procedures described above, since each requires the manual manipulation of all network activities between the subcontracted activity's longest predecessor and its longest successor. Fortunately, the majority of cases will be handled by the algorithms of Figures 9 and 10.

Appendix A contains sample networks of the type handled by each of the algorithmic devices and their solutions. Also included are the 10 cycle schedules determined manually for each of the networks. These are included to verify the correctness of the algorithmic solution.

The scheduling devices provided are limited in that they will promote work continuity for only a single activity. In Chapter V, there is some discussion of the problems involved in any extension of the system.

CHAPTER IV

CONCLUSIONS

The system derived is apparently capable of maximizing work continuity for a specific work activity in a cyclic project although no proof is currently available. It is readily adaptable to computer use and, as such, provides a means for determining the minimum number of work interruptions for a large cyclic network without working out the entire schedule. This might prove very useful in the planning phase of the project.

As previously mentioned, it could be very helpful for a contractor to have knowledge as to the minimum number of work interruptions for a particular subcontracted activity. With this information, he would be in a position to solicit two bids from potential subcontractors -- one based on the work to be done with no knowledge of scheduling to be used, and the other based on the knowledge that there will be a maximum of X interrupts. If the subcontractor's setup costs are high, the second bid might be considerably lower than the first. It is precisely cases of this type which will benefit most from a scheduling system such as this.

It should be realized that the results of obtaining a schedule which maximizes work continuity for a particular activity are not always desirable. Due to the interaction of all activities in a network, forcing the schedule to cater to the concerns of one activity causes changes in the schedule possibilities for the others which may not be acceptable. It will be up to the project planner to determine the cost effectiveness of maximizing continuity in an activity for his particular project.

Any or all of the following side effects may result in the course of maximizing work continuity for a particular activity.

1. The schedule possibilities for other activities may have less slack than they had before the application of the work continuity procedure.

2. The variability of daily resource levels will normally be greater than the optimal for the project.

3. The maximum daily resource usage will generally be greater than that called for by the best leveling schedule.

4. The subcontracted work has become critical and any delays will result in a greater project length.

These side effects are described in greater detail in Appendix B. As mentioned previously, they may or may not be significant in a particular project. It will be necessary for the individual planning a project to determine whether the savings derived from maximizing work continuity in an activity outweigh the increased costs which may result from the use of such a schedule.

Very little can be said about the cost-effectiveness in a general case except to point out that the more extensive the setup costs are for an activity, the greater will be the concern for work continuity. Cost trade-offs will be involved for any particular project and should be investigated.

CHAPTER V

RECOMMENDATIONS

This scheduling system should be extended to handle more than one activity. The interactions of the various activities creates considerable difficulties when considering extension of the single activity procedures. There would appear to be two ways of approaching the problem.

First, one might attempt to maximize the work continuity by stages. This might be done by using the system prescribed in this study to fix the schedule of one activity and then maximizing the work continuity of a second activity based on the fixed schedule for the first. A third activity could then be considered based on fixed schedules for the first two, etc. By following a procedure of this type, we would get an apparently optimal solution for the first activity considered, but those that followed would be less than optimal owing to the fact that their solution was constrained by the schedules which were fixed for previously considered activities.

A second approach to the problem might be to attempt to minimize the total number of interruptions for all activities considered. This approach would seem to be considerably more difficult in that there would be many more cases to consider and activity interrelationships would cause more problems. In dealing with multiple activities, it would be conceivable that we might have some preceding and others succeeding the pacing activity. Furthermore, each of the activities under consideration might have a different longest predecessor (or successor) activity. The possibilities appear endless and no method of solving this problem has been determined.

Another shortcoming of the second approach is that the minimization of the total number of interruptions will not always be a least-cost solution. The cost of interruption may vary considerably between activities and, conceivably, two interruptions of one activity might cost less than one interruption of another. It would be best to modify this approach so as to minimize the total cost of interruption rather than the total number of interruptions.

It seems that the first approach offers the greatest possibilities for development. Even though it will not minimize the total number of work interruptions or the total cost of interruption, it would be possible to treat activities in order of their costs of interruption and thus insure that the most critical activities (in terms of interruption costs) were scheduled subject to the fewest constraints. This would tend toward a least-cost solution even though it would be suboptimal.

APPENDICES

APPENDIX A

EXAMPLES OF THE USE OF ALGORITHMS

Example 1: (Using the network of Figure 8)

Since the subcontracted activity precedes the pacing activity, the algorithm of Figure 9 is used.

Input

A = 3	E = 1
B = 2	
C = 4	

Output

1st Output	L = 2	Q = 5	Z = 2	K = 0
2nd Output	L = 6	Q = 13	Z = 4	K = 1
3rd Output	L = 14	Q = 29	Z = 8	K = 2

Discussion of Results

Since $L_1 = 2$, it is possible to schedule two cycles with no interruptions ($K_1 = 0$). If 3 to 6 cycles are scheduled, there will be one interruption ($K_2 = 1$), and if 7 to 14 cycles are scheduled, two delays are necessary ($K_3 = 2$). Assuming that 10 cycles are desired, two cycles should be performed beginning at day 5 ($Q_1 = 5$, $Z_1 = 2$), 4 cycles should be performed beginning at day 13 ($Q_2 = 13$, $Z_2 = 4$), and the other four cycles should be performed beginning at day 29 ($Q_3 = 29$, $Z_3 > 4$). This is precisely the schedule shown in Figure 10.

Example 2

This is a ten cycle project with precedence relationships as shown in Figure 13. In this project, the pacing activity precedes the subcontracted activity and the subcontracted activity is not longer in duration than each of its successors. The algorithm of Figure 10 is used.

Input

N = 10	C = 4
A = 2	W = 3
B = 1	D = 2

Output

1st Output	L = 7	Q = 25	Z = 7	K = 0
2nd Output	L = 9	Q = 39	Z = 2	K = 1
3rd Output	L = 10	Q = 43	Z = 1	K = 2

The output corresponds to the schedule shown in Figure 14. Seven cycles of the subcontracted activity are begun on day 25 ($Q_1 = 25$, $Z_1 = 7$), two cycles are begun on day 39 ($Q_2 = 39$, $Z_2 = 2$), and one cycle is begun on day 43 ($Q_3 = 43$, $Z_3 = 1$).

Example 3

This is a ten cycle project with precedence relationships as shown in Figure 15. In this project, the pacing activity precedes the subcontracted activity, and the subcontracted activity is longer in duration than each of its predecessors. The procedural method of Figure 11 is used.

N = 10 B = 2 D = 1

$$(N-1)\ B-D-1 = 16$$

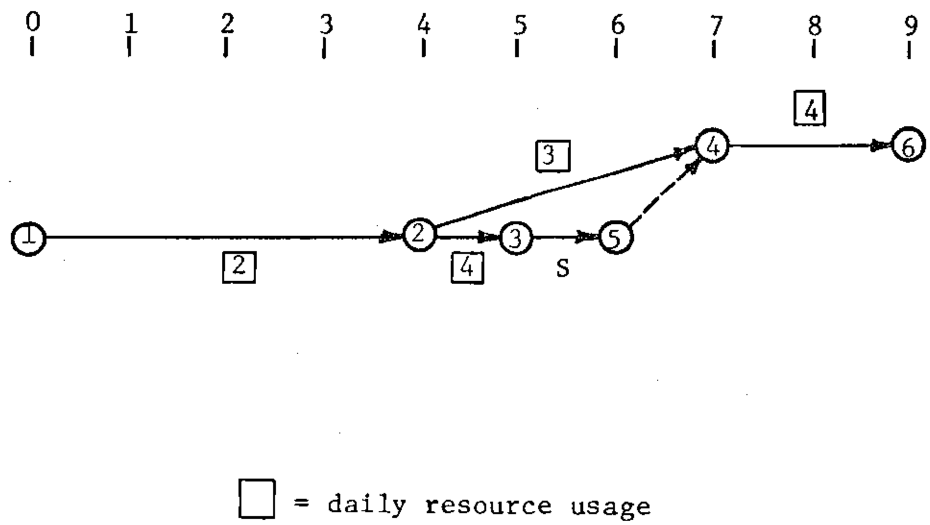
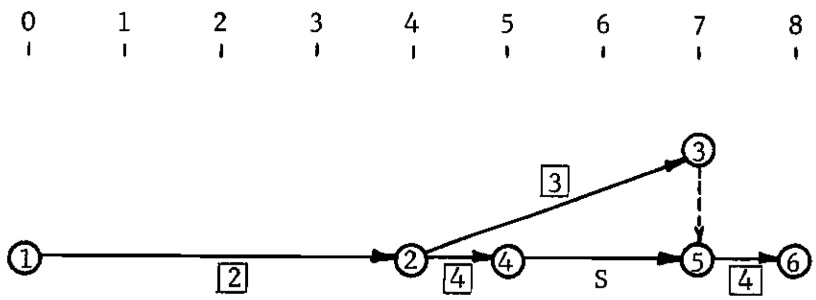


Figure 13. A Time Scaled Network Diagram for One Cycle of a Cyclic Project

					5					10					15					20		
1-2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2-3					4				4				4				4				4	
2-4					3	3	3		3	3	3		3	3	3		3	3	3		3	3
3-5																						
4-6																						

			25					30					35					40					45
1-2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2					
2-3			4				4				4				4				4				
2-4	3		3	3	3		3	3	3		3	3	3		3	3	3		3	3	3		
3-5			S	S	S	S	S	S	S								S	S			S		
4-6				4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4

Figure 14. A Schedule Allowing for Maximum Continuity of Activity 3-5



□ = daily resource usage

S = subcontracted

Figure 15. A Time Scaled Network Diagram for One Cycle of a Cyclic Project

The resulting schedule is shown in Figure 16. Notice that the subcontracted activity's predecessors have been placed at their earliest start, and its successors, at their latest start. The ten cycles of the subcontracted activity were begun at day 24 which is (N-1) B-D-1 = 16 days before the completion of the pacing activity.

Example 4

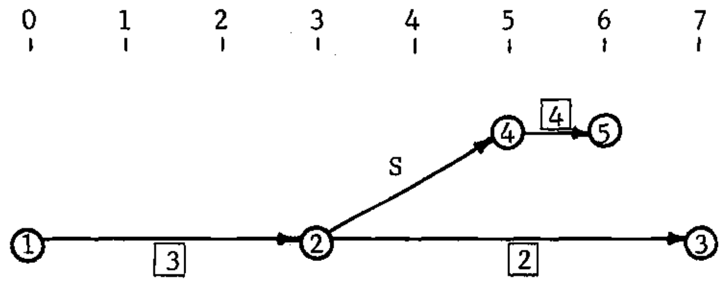
This is a ten cycle project with precedence relationships as shown in Figure 17. In this project the pacing activity neither precedes nor succeeds the subcontracted activity. The procedural method of Figure 12 is used.

The schedule which results is shown in Figure 18. The application of the procedural method is self explanatory.

					5					10					15					20		
1-2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2-3					3	3	3		3	3	3		3	3	3		3	3	3		3	3
2-4																						
4-5																						
5-6																						

			25					30					35					40					45
1-2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2					
2-3	3		3	3	3		3	3	3		3	3	3		3	3	3		3	3	3		
2-4	4		4		4		4		4		4		4		4		4		4				
4-5		S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S		
5-6				4		4		4		4		4		4		4		4		4		4	

Figure 16. A Schedule Allowing for Maximum Continuity of Activity 4-5



□ = daily resource usage

S = subcontracted

Figure 17. A Time Scaled Network Diagram for One Cycle of a Cyclic Project

					5					10					15					20		
1-2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
2-3				2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2-4																						S
4-5																						

			25					30					35					40					45
1-2	3	3	3	3	3	3	3	3															
2-3	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2		
2-4	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S				
4-5												4	4	4	4	4	4	4	4	4	4		

Figure 18. A Schedule Allowing for Maximum Continuity of Activity 2-4

APPENDIX B

DISCUSSION OF THE EFFECTS OF SCHEDULING FOR MAXIMUM CONTINUITY OF A SUBCONTRACTED ACTIVITY

The savings which may result from minimizing work interruptions have already been discussed in some detail. It must be pointed out that the total project cost will not always be decreased when this type of scheduling is used.

It is difficult to speak of project costs in general terms since the costs involved in one project may be of a completely different type from those involved in another. Normally, we would attempt to level daily resource requirements for a project as one means of minimizing costs. In other words, we would tend to associate some cost to the requirement for a varying manpower requirement. This cost might be a result of the administrative costs of hiring and firing or it might be merely the cost of transportation to the job site. Suffice it to say that a schedule with better resource leveling will normally be cheaper if all other things are equal.

The sums of squares of daily resource requirements is an indicator of the relative leveling of two schedules since it is a measure of variance. If two schedules having the same total length and the same total resource requirements are compared for relative level of resources, the one with the smallest sum of squares will have the best leveling -- its variation in resource requirements is the least.

Refer back to Figures 4 and 5 and recall that the schedule given in Figure 4 is that which minimized the sum of squares of daily resource requirements for the network shown. Figure 5 represents a schedule with maximum continuity of the subcontracted activity (2-3). The sum of squares for the schedule in Figure 4 is 2918 while that for the schedule in Figure 5 is 2972. This indicates that the maximum continuity schedule has greater variation in daily resource requirements. A maximum continuity schedule will not always result in an increased sum of squares, but in many cases it will. In the costing of a project, the possibility should be considered.

When the variation in resource requirements is increased, we often will increase the maximum daily resource requirement also. For instance, the schedule shown in Figure 4 required a maximum of nine resource units on any particular day while the maximum continuity schedule requires 12 resource units on days 18 and 19. If we were limited to 10 resource units this would obviously be an unacceptable schedule.

Another effect which we get from maximum continuity scheduling is the delaying of all subcontracted work until its latest start. It thus becomes a very critical part of the project schedule since any delays will cause the entire project to be delayed. A project manager may not always be willing to put this much faith in the performance of a subcontractor. If the subcontracted activity is subject to delays due to weather, strikes, etc., the project manager may not wish to risk delaying the project no matter how trustworthy the subcontractor.

By fixing the schedule for the subcontracted work, we also affect the scheduling possibilities for other activities in the network. In general, the start of predecessor activities will be pushed towards their earliest start and the start of successor activities will be pushed towards their latest start.

Of course, all of these effects must be considered for each individual project. A cost-effectiveness appraisal must be made by management in order to determine whether the maximizing of work continuity is justified in a particular project.

BIBLIOGRAPHY

1. Bennet, F. L., Some Approaches to the Critical Path Scheduling Resource Allocation Problem, Ph.D. Thesis, Cornell University, Ithaca, New York, 1966.

2. Berman, E. B., "Resource Allocation in a PERT Network Under Continuous Activity Time Cost Functions," Management Science, Vol. 10, No. 4 (1964), pp. 734-745.

3. Burgess, A. R., and Killebrew, J. B., "Variation in Activity Level on a Cyclical Arrow Diagram," Journal of Industrial Engineering, Vol. 13, No. 2 (1962), pp. 76-83.

4. Carruthers, J. A., and Battersby, A., "Advances in Critical Path Methods," Operations Research Quarterly, Vol. 17, No. 1 (1966), pp. 359-380.

5. Cleland, D. I., Systems Analysis and Project Management, McGraw-Hill Book Company, Inc., New York, 1968, 315 pages.

6. Crowstone, W. B., and Thompson, G. L., "Decision CPM: A Method for Simultaneous Planning, Scheduling, and Control of Projects," Operations Research, Vol. 5, No. 3 (1967), pp. 407-426.

7. Davis, E. W., "Resource Allocation in Project Network Models -- A Survey," Journal of Industrial Engineering, Vol. 17, No. 4 (1966), pp. 177-188.

8. Davis, J. G., and Buchan, J. R., "The REST Approach to Scheduling with Limited Resources," Journal of Systems Management, Vol. 20, No. 8 (1969), pp. 7-11.

9. Fisher, C., and Nemhauser, G. L., "Multicycle Project Planning," Journal of Industrial Engineering, Vol. 18, No. 4 (1967), pp. 278-284.

10. Jewell, W. S., "Risk-taking in Critical Path Analysis," Management Science, Vol. 11, No. 3 (1965), pp. 438-443.

11. Johnson, T. J. R., An Algorithm for the Resource Constrained Project Scheduling Problem, Unpublished Ph.D. Thesis, Sloan School of Manufacturing, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1967.

BIBLIOGRAPHY (Continued)

12. Klein, M. M., "Scheduling Project Networks," Communication of the ACM, Vol. 10 (1967), pp. 225-231.

13. Lambourn, S., "Resource Allocation and Multi-Project Scheduling (RAMPS) -- A New Tool in Planning and Control," Computer Journal, Vol. 5 (1963), pp. 300-304.

14. Levy, F. K., Thompson, G. L., and Wiest, J. D., "Multi-ship, Multi-shop Workload Smoothing Program," Naval Research Logistics Quarterly, Vol. 9, No. 1 (1962), pp. 37-44.

15. McNeill, C. A., Development of a Resource Allocation Technique Based on Resource Time-Units, M. S. Thesis, Georgia Institute of Technology, Atlanta, Georgia, 1967.

16. Martino, R. L., Resources Management, Management Development Institute, Wayne, Pennsylvania, 1968, 147 pages.

17. Meyer, W. L., and Shaffer, L. R., "Extending CPM for Multiform Project Time-Cost Curves," Journal of the Construction Division, Proceedings of the American Society of Civil Engineers, Vol. 91, No. CO14319 (1965), pp. 45-67.

18. Moder, J. J., and Phillips, C. R., Project Management with CPM and PERT, Reinhold Publishing Corporation, New York, 1966, 283 pages.

19. Moshman, J., Johnson, J., and Larsen, M., "RAMPS, A Technique for Resource Allocation and Multiproject Scheduling," Proceedings, 1963 Spring Joint Computer Conference, Vol. 23 (1963), pp. 17-27.

20. Petrović, R., "Optimization of Resource Allocation in Project Planning," Operations Research, Vol. 16 (1968), pp. 559-568.

21. Pritsker, A. A. B., and Watters, L. J., "A Zero-One Programming Approach to Scheduling with Limited Resources," The RAND Corporation, Memorandum RM-5561-PR, January 1968.

22. Pritsker, A. A. B., Watters, L. J., and Wolfe, P. M., "Multiproject Scheduling with Limited Resources: A Zero-One Programming Approach," Management Science, Vol. 16, No. 1 (1969), pp. 93-108.

23. Rosenbloom, R. S., "Notes on the Development of Network Models for Resource Allocation in R & D Projects," IEEE Transactions on Engineering Management, Vol. EM 11 (1964), pp. 58-63.

BIBLIOGRAPHY (Concluded)

24. Weist, J. D., "A Heuristic Model for Scheduling Large Projects with Limited Resources," Management Science, Vol. 13, No. 6 (1967), pp. B-359-377.

25. Weist, J. D., "Some Properties of Schedules for Large Projects with Limited Resources," Operations Research, Vol. 12 (1964), pp. 395-418.

26. Weist, J. D., The Scheduling of Large Projects with Limited Resources, Ph.D. Dissertation, Carnegie Institute of Technology, Pittsburgh, Pennsylvania, 1963.

27. Wilson, R. C., "Assembly Line Balancing and Resource Leveling," University of Michigan Summer Conference, Production and Inventory Control, 1964.

28. Wolfe, P. M., A Zero-One Programming Approach to Multiproject Scheduling with Limited Resources, Ph.D. Thesis, Arizona State University, Tempe, Arizona, 1968.

29. Woodgate, H. S., Planning by Network: Project Planning and Control Using Network Techniques, Business Publications, London, 1967, 385 pages.

30. RAMPS Users Guide, C.E.I.R. Inc., Washington, D. C., 1963.